National Aeronautics and Space Administration

Using SciCon Tools to Help with Productivity

NAS Webinar

Aug 24, 2016

NASA Advanced Supercomputing Division

www.nasa.gov

Using SciCon Tools to Help with Productivity

Abstract:

There are multiple tools created by the SciCon members over the years to help users with various tasks. An overview of these tools will be provided in this webinar where users can learn how to check available licenses, status of PBS resources, status of user's PBS jobs, amount of memory used in a job, debug a running job, pin processes for better performance, and analyze performance for MPI applications.

Question? Use the Webex chat facility to ask the Host

List of Tools

- `check_licenses`: Availability of Licenses for 3rd Party Tools
- `node_stats`: Overview Node Usage in PBS
- `qs`: An Abstract View of the PBS Queues
- `qsh.pl`: Run Commands on All Nodes Inside of PBS Job
- `qtop.pl`: Run Top Across All Nodes Inside of PBS Job
- `gm.x`: Memory Allocated Per Rank Inside a PBS Job
- `mbind.x`: Bind Processes to Specific Cores on A Node
- `pdsh_gdb`: Obtain Stacktrace of Running Job
- *`myNAS`*: A Mobile App for Monitoring PBS Jobs & Resources
- `mpiprof`: MPI & IO Performance Monitoring Tool

check_licenses

Question? Use the Webex chat facility to ask the Host

check_licenses

- /u/scicon/tools/bin/check_licenses
  - This tool allows a user to determine how many licenses are being used for a number of 3rd party tools installed on Pleiades depending on which command line argument is given.
  - Example: /u/scicon/tools/bin/check_licenses -m

```
Users of MATLAB:  (Total of 16 licenses issued;  Total of 14 licenses in use)
          1 RESERVATION for GROUP xxxxx (server/1705)
    user1 host9 /dev/pts/4 (v28) (server/1705 1202), start Thu 6/16 8:13
    user2 host6 /dev/pts/15 (v32) (server /1705 2008), start Thu 6/16 10:40
    user3 host1 /dev/pts/0 (v28) (server/1705 1605), start Thu 6/16 6:27
    user4 host3100 (v32) (server/1705 1908), start Thu 6/16 10:41
    user5 host5 /dev/pts/55 (v32) (server/1705 703), start Thu 6/16 8:06
    user6 host8 /dev/pts/96 (v30) (server/1705 401), start Tue 6/14 7:47
    user7 host2 /dev/pts/17 (v30) (server/1705 302), start Wed 6/15 10:45
    user8 host7 /dev/pts/9 (v32) (server/1705 207), start Thu 6/16 3:30
    user9 host6 /dev/pts/79 (v32) (server/1705 102), start Tue 6/14 7:44
    user10 host4 /dev/pts/50 (v30) (server/1705 1002), start Wed 6/15 8:41
```

check_license

```
Usage: /u/scicon/tools/bin/check_licenses 'flag'
        where 'flag' is one of -h -m -M -i -p -g -t -f -e -l -s
          -h : gives this help
          -m : lists basic Matlab license
          -M : lists Matlab and all its Toolkits
          -c : lists tecplot licenses
          -e : lists EXA licenses  (limited access)
          -i : lists IDL licenses
          -l : lists Celeritas licenses  (limited access)
          -p : lists Pointwise licenses
          -g : lists GridGen licenses
          -s : lists StarCCM licenses  (limited access)
          -t : lists Totalview licenses
          -f : lists Fluent licenses  (limited access)
          -v : lists Fieldview licenses
          -n : lists Intel Compiler licenses
          -w : lists Flow3d licenses (limited access)
```

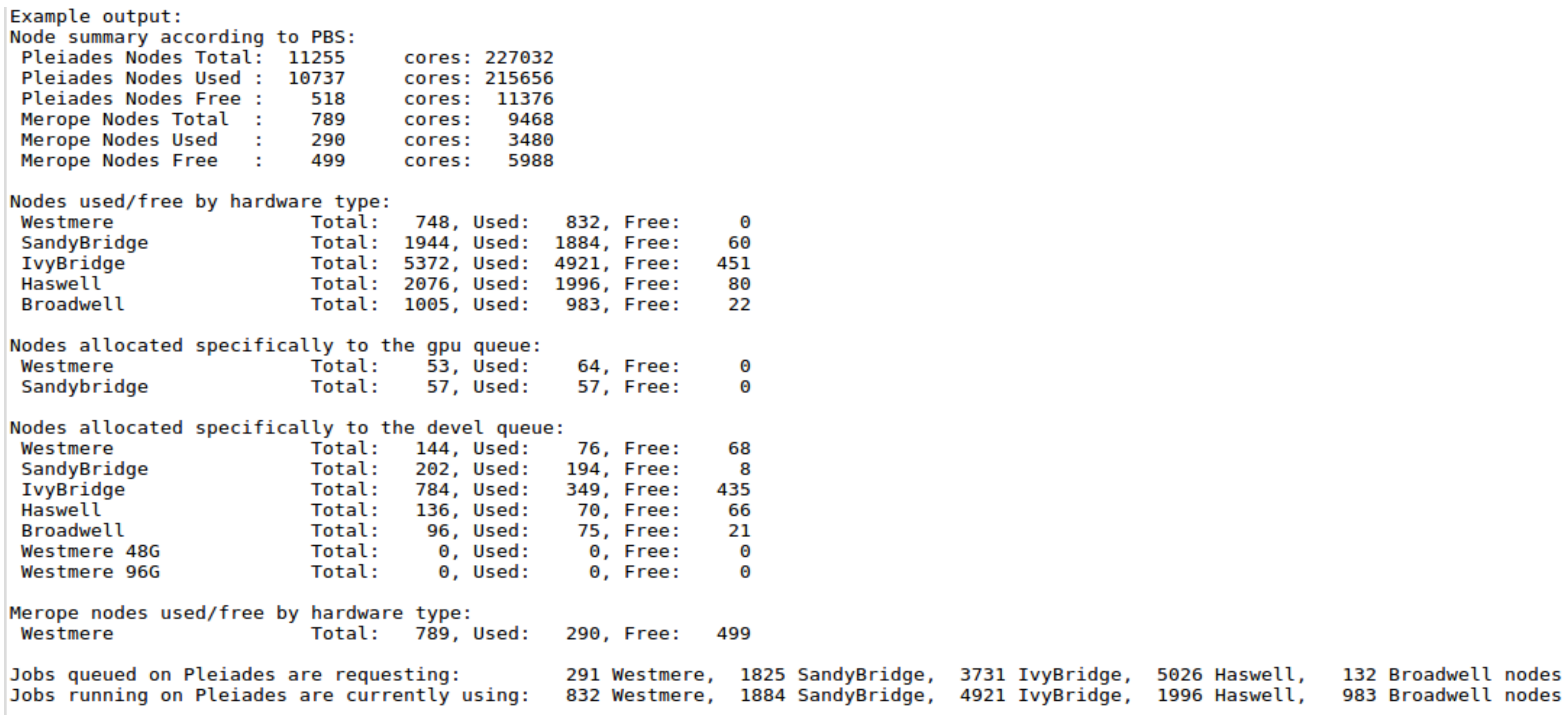
Question? Use the Webex chat facility to ask the Host

node_stats

Question? Use the Webex chat facility to ask the Host

node_stats

- /u/scicon/tools/bin/node_stats
  - Will display a snapshop of the state of the various node types/queues/ PBS jobs on Pleaides.



```
Example output:
Node summary according to PBS:
 Pleiades Nodes Total:  11255     cores: 227032
 Pleiades Nodes Used :  10737     cores: 215656
 Pleiades Nodes Free :    518     cores:  11376
 Merope Nodes Total  :    789     cores:   9468
 Merope Nodes Used   :    290     cores:   3480
 Merope Nodes Free   :    499     cores:   5988

Nodes used/free by hardware type:
 Westmere                 Total:   748, Used:   832, Free:     0
 SandyBridge              Total:  1944, Used:  1884, Free:    60
 IvyBridge                Total:  5372, Used:  4921, Free:   451
 Haswell                  Total:  2076, Used:  1996, Free:    80
 Broadwell                Total:  1005, Used:   983, Free:    22

Nodes allocated specifically to the gpu queue:
 Westmere                 Total:    53, Used:    64, Free:     0
 Sandybridge              Total:    57, Used:    57, Free:     0

Nodes allocated specifically to the devel queue:
 Westmere                 Total:   144, Used:    76, Free:    68
 SandyBridge              Total:   202, Used:   194, Free:     8
 IvyBridge                Total:   784, Used:   349, Free:   435
 Haswell                  Total:   136, Used:    70, Free:    66
 Broadwell                Total:    96, Used:    75, Free:    21
 Westmere 48G             Total:     0, Used:     0, Free:     0
 Westmere 96G             Total:     0, Used:     0, Free:     0

Merope nodes used/free by hardware type:
 Westmere                 Total:   789, Used:   290, Free:   499

Jobs queued on Pleiades are requesting:       291 Westmere,  1825 SandyBridge,  3731 IvyBridge,  5026 Haswell,   132 Broadwell nodes
Jobs running on Pleiades are currently using:  832 Westmere,  1884 SandyBridge,  4921 IvyBridge,  1996 Haswell,   983 Broadwell nodes
```

Question? Use the Webex chat facility to ask the Host

`qs`: A More Abstract View of What’s Going on in the Queues

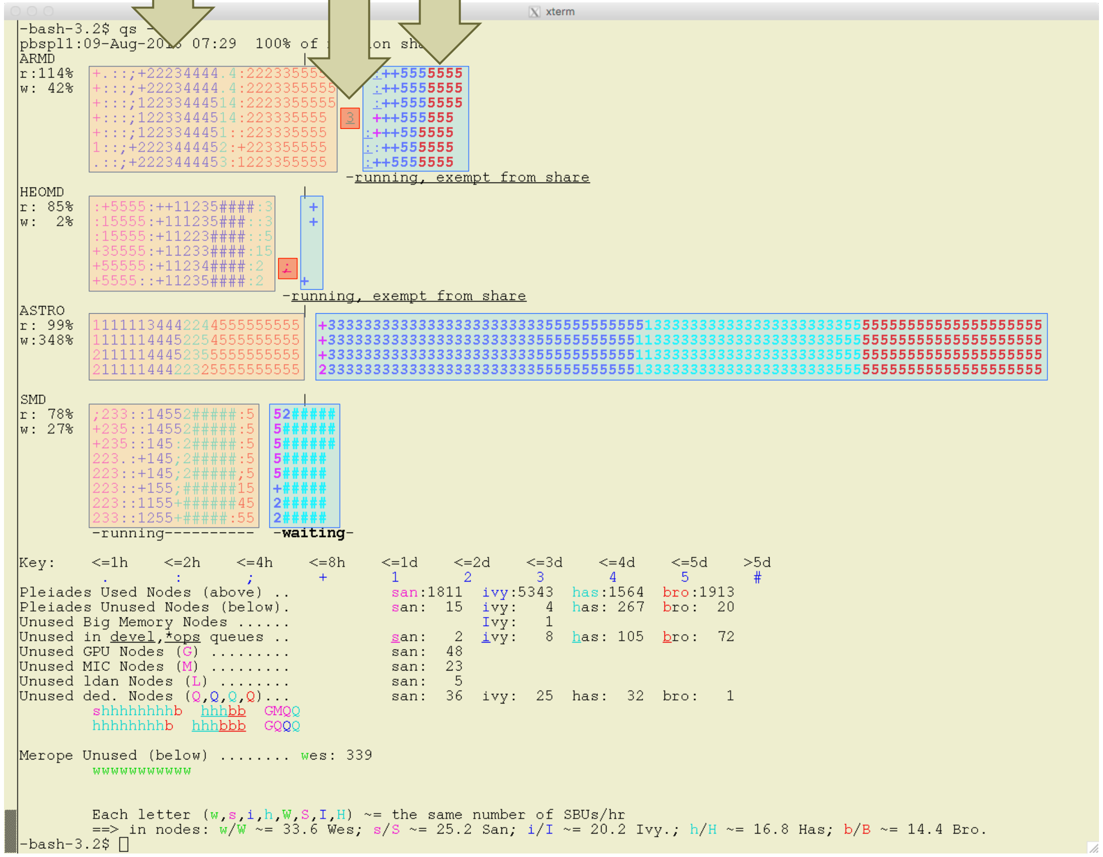
Question? Use the Webex chat facility to ask the Host

Use qs to Get an Overview of Resources

Resources being used as part of a Mission's share

Resources being used, but not counted against Mission's share

Resources requested by jobs in the queue



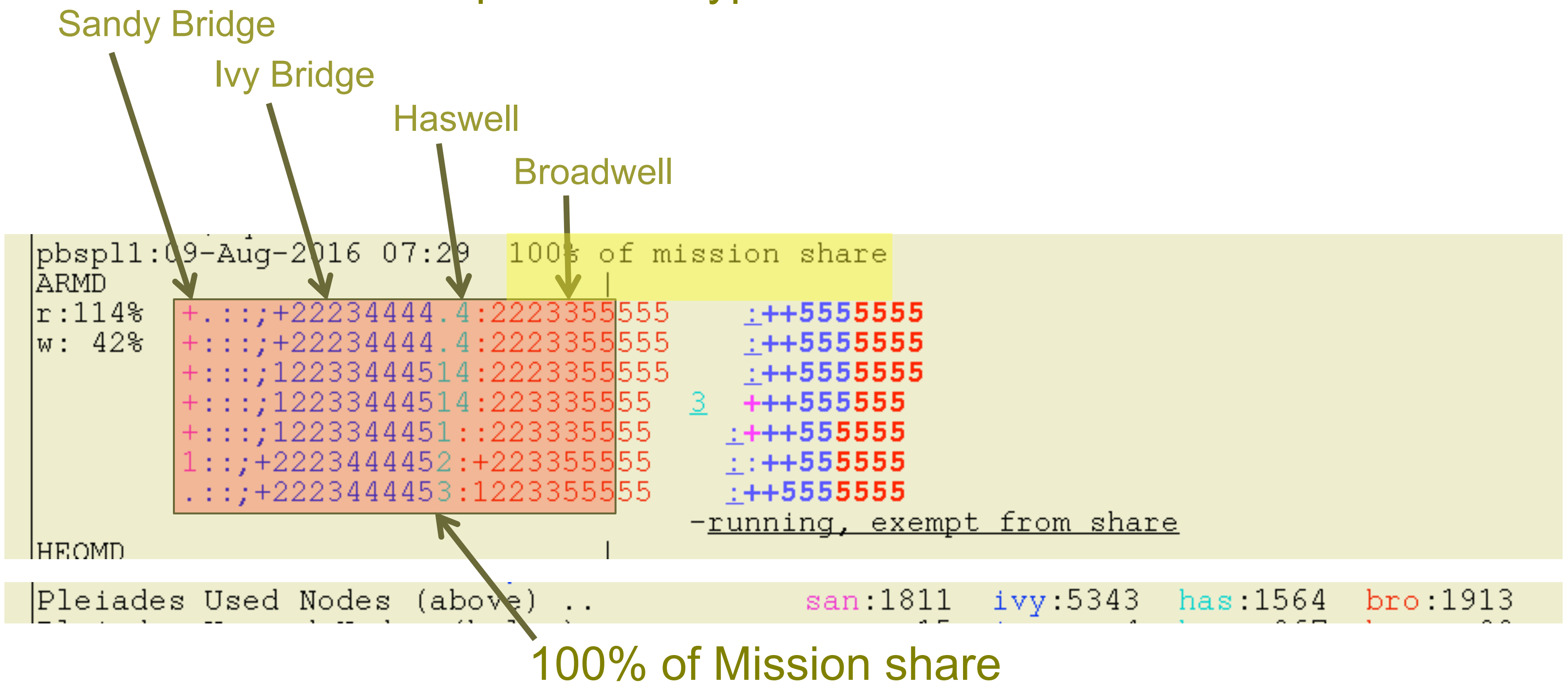
```
-bash-3.2$ qs -
pbspl1:09-Aug-201  07:29  100% of    on sha
ARMD                                    |
r:114%   +.:::;+22234444.4:222335555     :++5555555
w: 42%   +:::;+22234444.4:2223355555     :++5555555
         +:::;1223344451 4:2223355555    :++5555555
         +:::;1223344451 4:223335555  3  +++555555
         +:::;122334445 1::223335555     :+++55555
         1::;+222344445 2:+223355555     ::++55555
         .::;+222344445 3:1223355555     :++555555
                                       -running, exempt from share
HEOMD                          |
r: 85%   :+5555:++11235####:3     +
w:  2%   :15555:+111235###::3     +
         :15555:+11223####::5
         +35555:+11233####:15
         +55555:+11234####:2   ;
         +5555::+11235####:2      +
                               -running, exempt from share
ASTRO                           |
r: 99%   1111113444224555555555   +33333333333333333333333355555555555133333333333333333333355555555555555555555
w:348%   1111114445225455555555   +333333333333333333333333555555555551133333333333333333333555555555555555555555
         2111114445235555555555   +333333333333333333333333555555555551133333333333333333333555555555555555555555
         2111114442232555555555   2333333333333333333333333555555555551333333333333333333333555555555555555555555

SMD                             |
r: 78%   ;233::14552#####:5   52#####
w: 27%   +235::14552#####:5   5######
         +235::145:2#####:5   5######
         223.:+145;2#####:5   5#####
         223::+145;2#####;5   5#####
         223::+155;######15   +#####
         223::1155+######45   2#####
         233::1255+#####:55   2#####
         -running----------   -waiting-

Key:     <=1h     <=2h     <=4h     <=8h     <=1d     <=2d     <=3d     <=4d     <=5d     >5d
           .        :        ;        +        1        2        3        4        5        #
Pleiades Used Nodes (above) ..                  san:1811  ivy:5343  has:1564  bro:1913
Pleiades Unused Nodes (below).                  san:  15  ivy:   4  has: 267  bro:  20
Unused Big Memory Nodes ......                            Ivy:   1
Unused in devel,*ops queues ..                  san:   2  ivy:   8  has: 105  bro:  72
Unused GPU Nodes (G) .........                  san:  48
Unused MIC Nodes (M) .........                  san:  23
Unused ldan Nodes (L) ........                  san:   5
Unused ded. Nodes (Q,Q,Q,Q)...                  san:  36  ivy:  25  has:  32  bro:   1
         shhhhhhhhb  hhhbb  GMQQ
         hhhhhhhhb  hhhbbb  GQQQ

Merope Unused (below) ........ wes: 339
         wwwwwwwwwww


         Each letter (w,s,i,h,W,S,I,H) ~= the same number of SBUs/hr
         ==> in nodes: w/W ~= 33.6 Wes; s/S ~= 25.2 San; i/I ~= 20.2 Ivy.; h/H ~= 16.8 Has; b/B ~= 14.4 Bro.
-bash-3.2$
```

Question? Use the Webex chat facility to ask the Host

What a Mission is Doing

Each symbol represents the same number of SBU's, either in use or requested

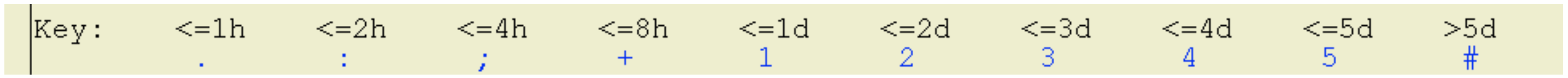
Colors are used to indicate processor type:



```
pbspl1:09-Aug-2016 07:29   100% of mission share
ARMD                                 |
r:114%   +.:::;+22234444.4:2223355555       :++5555555
w:  42%  +::::;+22234444.4:2223355555       :++5555555
         +::::;12233444514:2223355555       :++5555555
         +::::;12233444514:223335555   3    +++555555
         +::::;122334445l::223335555       :+++555555
         1::;+2223444452:+223355555        ::++555555
         .::;+2223444453:122335555         :++5555555
                                          -running, exempt from share
HEOMD                                |
```

```
Pleiades Used Nodes (above) ..             san:1811  ivy:5343  has:1564  bro:1913
```

Symbol depicts time remaining/requested for job



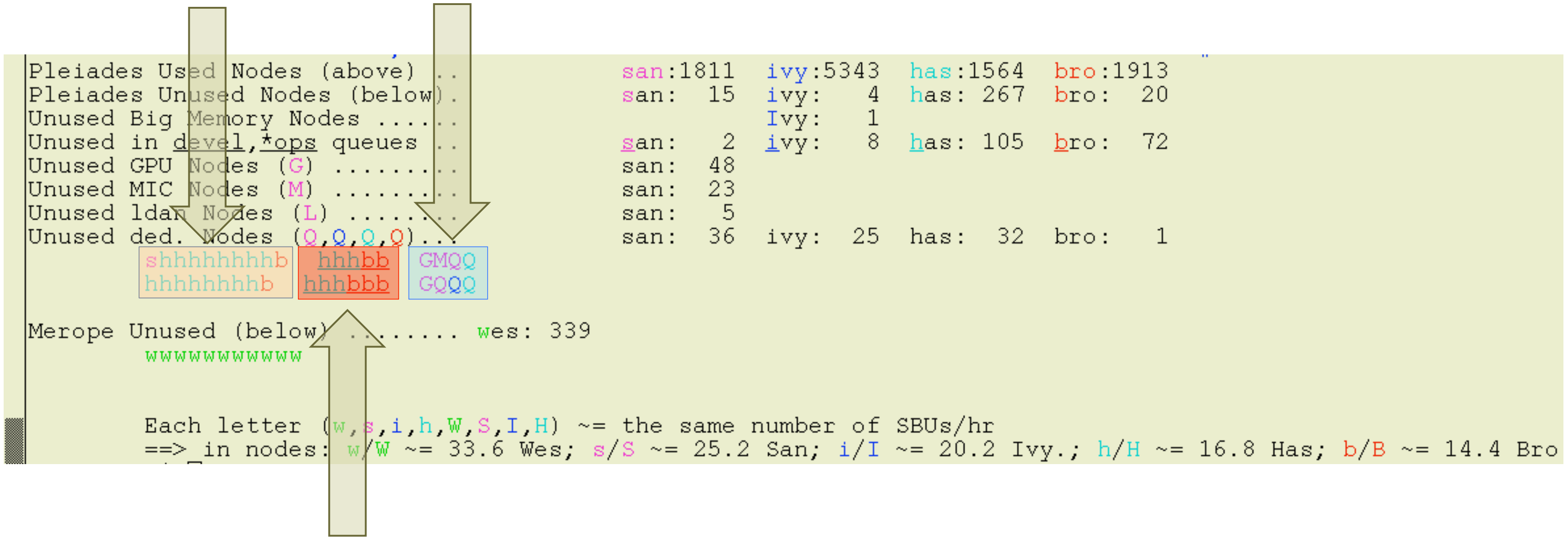
```
Key:    <=1h   <=2h   <=4h   <=8h   <=1d   <=2d   <=3d   <=4d   <=5d   >5d
         .      :      ;      +      1      2      3      4      5      #
```

Question? Use the Webex chat facility to ask the Host

Free/Available Resources

Available Resources

Unused Resources in Dedicated Queues



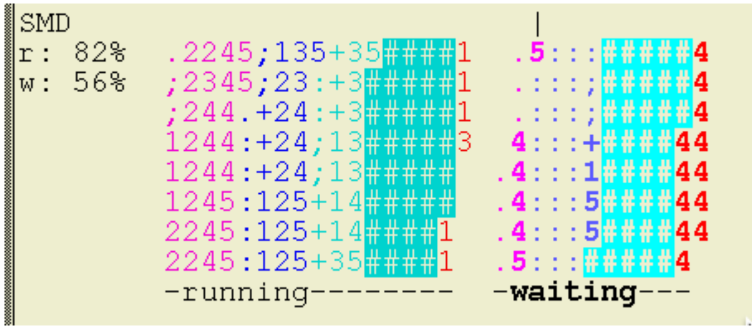
```
Pleiades Used Nodes (above) ..           san:1811  ivy:5343  has:1564  bro:1913
Pleiades Unused Nodes (below).           san:  15  ivy:   4  has: 267  bro:  20
Unused Big Memory Nodes .......                    Ivy:   1
Unused in devel,*ops queues ..           san:   2  ivy:   8  has: 105  bro:  72
Unused GPU Nodes (G) .........           san:  48
Unused MIC Nodes (M) .........           san:  23
Unused ldan Nodes (L) ........           san:   5
Unused ded. Nodes (Q,Q,Q,Q)...           san:  36  ivy:  25  has:  32  bro:   1
        shhhhhhhhb  hhhbb  GMQQ
        hhhhhhhhb  hhhbbb  GQQQ

Merope Unused (below) ........ wes: 339
        wwwwwwwwwww


        Each letter (w,s,i,h,W,S,I,H) ~= the same number of SBUs/hr
        ==> in nodes: w/W ~= 33.6 Wes; s/S ~= 25.2 San; i/I ~= 20.2 Ivy.; h/H ~= 16.8 Has; b/B ~= 14.4 Bro
```

Unused Resources in “devel” and other demand queues

Question? Use the Webex chat facility to ask the Host

Useful Options

- `-d` (for a light background) use dark letters
- `-u` highlight my jobs



```
SMD                               |
r:  82%   .2245;135+35####1    .5:::#####4
w:  56%   ;2345;23:+3#####1    .:::;#####4
          ;244.+24:+3#####1    .:::;#####4
          1244:+24;13#####3     4:::+####44
          1244:+24;13#####     .4:::1####44
          1245:125+14#####     .4:::5####44
          2245:125+14####1     .4:::5####44
          2245:125+35####1     .5:::#####4
          -running--------    -waiting---
```

- `-w` show time spent waiting for resources instead of time remaining/requested
- `-v` show explanatory output
- `-n` *`M`* show time remaining before *`M`* nodes are free (ignores queued jobs, however)

Question? Use the Webex chat facility to ask the Host

qsh.pl

Question? Use the Webex chat facility to ask the Host

qsh.pl

- /u/scicon/tools/bin/qsh.pl PBS_JOBID ‘command to run’
  - Will run a command across all the nodes in a user’s running job.

```
/u/scicon/tools/bin/qsh.pl 123456 ‘ps axuwlgrep a.out’

/u/scicon/tools/bin/qsh.pl 123456 ‘grep MemFree /proc/meminfo’

/u/scicon/tools/bin/qsh.pl 123456
‘/bin/ps -A -o psr,%cpu,comm,ppid,pid,rss,size,user,etime,cputime l grep
username l grep a.out l sort –n’
```

Question? Use the Webex chat facility to ask the Host

qtop.pl

qtop

/u/scicon/tools/bin/qtop.pl

```
usage:   qtop.pl [-b] [-p n] [-P s] [-h n] [-H s] [-t s] [-N s] PBSjobid
           -b   : (for running in background or batch) don't run 'resize' command
           -p n : show at most n processes per host
           -P s : show only procs in s, a comma-separated list of ranges
                  e.g. -P 1,8-9
           -h   : don't show the column header line
           -H s : show only header lines in s, comma-separated ranges
                  e.g. -H 1-2,7
                  e.g. -H 0 (don't show any lines)
           -t s : pass string s (must be one argument) to top command
           -n s : show output only from nodes in s, comma-separated ranges
                  e.g. -n 0,2-3             (relative node #'s)
           -N s : show output only from nodes in s, a comma-separated list
                  e.g. -N r1i1n14,r1i1n15 (absolute node #'s)
```

Question? Use the Webex chat facility to ask the Host

gm.x – Report and Monitor Memory Usage of an Application

Question? Use the Webex chat facility to ask the Host

Summary of gm.x

- Types of memory usage
  - hwm – high water mark (upper bound)
  - rss – resident memory size (at a given instance)
- Usage syntax
  - Report memory usage at the end of a run

    ```
    mpiexec -np <n> gm.x [-type] a.out
    ```
  - Monitor memory usage during a run

    ```
    mpiexec -np <n> gm.x [-type] -c<s[:nc]> a.out
    ```
    - <s> is the interval in seconds, <nc> is the number of times
- Post processing
  - Report aggregated results (total, average, minimum, maximum)

    ```
    gmpost.x gm_output
    mpiexec -np <n> gm.x a.out | gmpost.x -v
    ```

Question? Use the Webex chat facility to ask the Host

gm.x – Examples

- Case 1
  - To report memory usage of a single process (serial code)

```
gm.x a.out
```

- Case 2
  - To report memory usage of an MPI code with 8 ranks

```
mpiexec -np 8 gm.x a.out
```

- Case 3
  - To monitor memory usage of an MPI job at every 10 seconds

```
mpiexec -np 8 gm.x -c10 a.out
```

- Additional information
  - Runtime help (`gm.x -help`)
  - Knowledge base article (220)

http://www.nas.nasa.gov/hecc/support/kb/checking-memory-usage-of-a-pleiades-or-endeavour-batch-job-with-gmx_220.html

Question? Use the Webex chat facility to ask the Host

mbind.x – Bind Processes and Threads to CPUs

Summary of mbind.x

- Purpose
  - Provide a uniform interface for binding processes and threads to CPUs for MPI, OpenMP, or MPI+OpenMP applications
  - Improve performance, especially in “sparse” use of resources
  - Reduce timing variation from run to run
- Supported libraries
  - MPI: SGI-MPT, Intel-MPI, Open-MPI, MVAPICH2, MPICH2
  - OpenMP: Intel OpenMP runtime, GNU OpenMP, PGI runtime
- Usage syntax
  - OpenMP codes

```
mbind.x -t<nt> a.out
```

  - MPI or MPI+OpenMP codes

```
mpiexec -np <n> mbind.x -n<npp> -t<nt> a.out
```

  - Notation

`<n>`	-	number of MPI processes
`<npp>`	-	number of MPI processes per node
`<nt>`	-	number of OpenMP threads per process

A Few Recommended mbind.x Options

`-cs`	spreads the processes and threads among physical CPUs to minimize resource contentions, and is the default placement
`-cp`	places the processes and threads as close as possible onto CPUs to improve shared resource performance
`-cc`	places the processes and threads cyclically onto CPUs from each socket to distribute resource utilization
`-c`*<cpulist>*	specifies an explicit list of CPUs for the processes and threads to be pinned (for example `–c`0,3,6,9). Use of [`–cs`l`-cp`l`-cc`] is preferred.
`-n`*<npp>*	sets the number of processes per node. Default is the maximum number of cores on a node.
`-t`*<nt>*	sets the number of threads per process. This option overrides the value specified by `OMP_NUM_THREADS`. Default is 1 if `OMP_NUM_THREADS` is not specified.
`-v`	sets the verbose flag, useful for verifying the binding
`-s`*<seq>*	specifies the type of the CPU sequence (*<seq>*=`core`l`smt`) used for binding. The "`core`" sequence (as the default) does not include hyperthreads; the "`smt`" sequence includes hyperthreads.

Question? Use the Webex chat facility to ask the Host

mbind.x – Examples

- Case 1
  - OpenMP code, spread 4 OpenMP threads on cores in a node

```
mbind.x -cs -t4 -v ./a.out
host: r211i0n5, ncpus 24, nthreads: 4, bound to cpus: {0,3,6,9}
```

- Case 2
  - MPI code, bind 32 MPI ranks to 2 nodes, with 16 ranks per node

```
mpiexec -np 32 mbind.x -n16 -v ./a.out
```

- Case 3
  - Hybrid MPI+OpenMP code for a total 8 MPI ranks and 4 OpenMP threads per rank across 2 nodes, 4 MPI ranks per node

```
mpiexec -np 8 mbind.x -n4 -t4 -v ./a.out
```

- Additional information
  - Runtime help (`mbind.x -help`)
  - Knowledge base article (288)

http://www.nas.nasa.gov/hecc/support/kb/using-the-mbind-tool-for-pinning_288.html

Question? Use the Webex chat facility to ask the Host

pdsh_gdb

Question? Use the Webex chat facility to ask the Host

pdsh_gdb

- /u/scicon/tools/bin/pdsh_gdb –j JOBID –s –d directory –n group_name
  - This will go out on all the nodes in the currently running job and run gdb on each user process to get a stack trace.
  - Note that if you had submitted the job using a group allocation that isnt the default you will need to give the -n option. eg. if -W group_list=X was used, then -n X will also need to be given to pdsh_gdb.
  - The -s option will print a summary as best it can across all the processes. It will group all the processes that were doing the same subroutine together. This can be useful in determining if a process was off somewhere unexpected compared to the others.
  - A full stack trace for each process is written to its own file inside the given directory.
  - If a run is hanging and you aren't sure why, this tool can be used to get the current state of all the processes which can show if one crashed or off in a state it shouldn't be.
  - Note that this tool will on occasion fail badly. In some cases which we havent figured out why yet, the gdb operation on the compute nodes will segfault and crash.

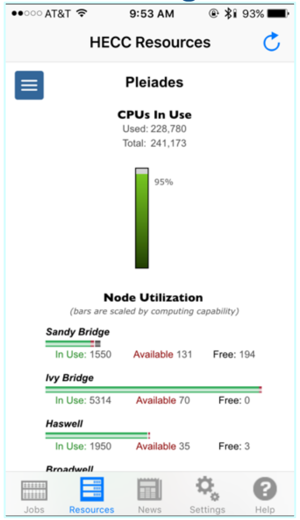
Question? Use the Webex chat facility to ask the Host

The *myNAS* Mobile App

Question? Use the Webex chat facility to ask the Host

Getting Job Information When You're Away from your Desk

- *myNAS* Features
  - Monitoring HECC Resources



HECC Resources

Pleiades

CPUs In Use
Used: 228,780
Total: 241,173
95%

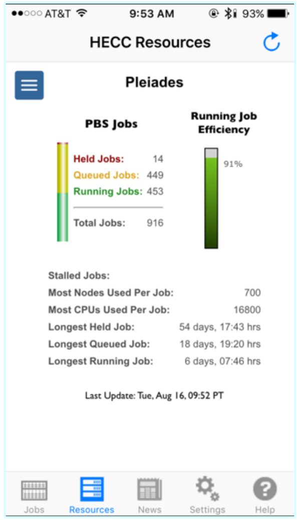
Node Utilization
(bars are scaled by computing capability)

Sandy Bridge
In Use: 1550 Available 131 Free: 194

Ivy Bridge
In Use: 5314 Available 70 Free: 0

Haswell
In Use: 1950 Available 35 Free: 3

Broadwell



HECC Resources

Pleiades

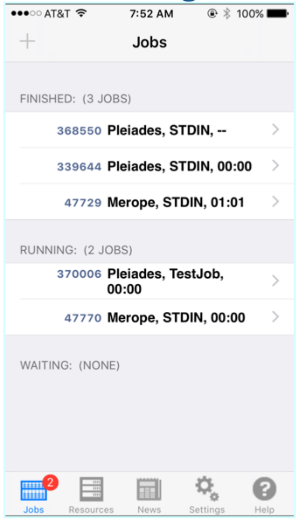
PBS Jobs
Held Jobs: 14
Queued Jobs: 449
Running Jobs: 453
Total Jobs: 916

Running Job Efficiency
91%

Stalled Jobs:	
Most Nodes Used Per Job:	700
Most CPUs Used Per Job:	16800
Longest Held Job:	54 days, 17:43 hrs
Longest Queued Job:	18 days, 19:20 hrs
Longest Running Job:	6 days, 07:46 hrs

Last Update: Tue, Aug 16, 09:52 PT

Monitoring PBS Jobs (including notifications)

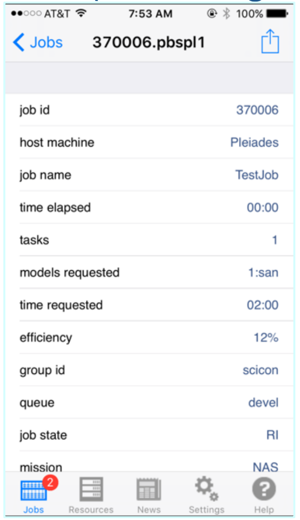


Jobs

FINISHED: (3 JOBS)
368550 Pleiades, STDIN, --
339644 Pleiades, STDIN, 00:00
47729 Merope, STDIN, 01:01

RUNNING: (2 JOBS)
370006 Pleiades, TestJob, 00:00
47770 Merope, STDIN, 00:00

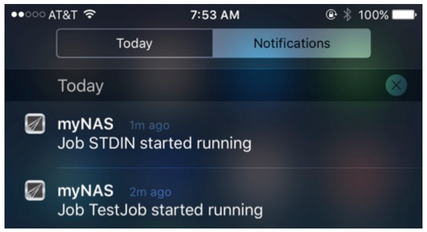
WAITING: (NONE)



370006.pbspl1

job id	370006
host machine	Pleiades
job name	TestJob
time elapsed	00:00
tasks	1
models requested	1:san
time requested	02:00
efficiency	12%
group id	scicon
queue	devel
job state	Rl
mission	NAS

  - Viewing output/error files
  - Notifications of job state changes



Today

myNAS 1m ago
Job STDIN started running

myNAS 2m ago
Job TestJob started running

- More info

  http://www.nas.nasa.gov/hecc/support/kb/installing-the-mynas-mobile-app_470.html

  http://www.nas.nasa.gov/hecc/support/kb/using-the-mynas-mobile-app_465.html

Question? Use the Webex chat facility to ask the Host

mpiprof – Analyze Performance for MPI Applications and Report I/O Statistics

For usage of mpiprof, please refer to

- Knowledge base article (525)

  http://www.nas.nasa.gov/hecc/support/kb/using-mpiprof-for-performance-analysis_525.html

- Presentation from the previous NAS webinar

  http://www.nas.nasa.gov/hecc/assets/pdf/training/Using_MPIprof_for_Performance_Analysis_2016_07_20.pdf

Question? Use the Webex chat facility to ask the Host